

**MODULATION DEVICE AND METHOD, DEMODULATION DEVICE AND METHOD AND PROVIDING MEDIUM**

**Publication Number:** 2000-068847 (JP 2000068847 A) , March 03, 2000

**Inventors:**

- NAKAGAWA TOSHIYUKI
- NIIFUKU YOSHIHIDE
- NARAHARA TATSUYA
- NAKAMURA KOSUKE

**Applicants**

- SONY CORP

**Application Number:** 10-237045 (JP 98237045) , August 24, 1998

**International Class:**

- H03M-007/14

**Abstract:**

**PROBLEM TO BE SOLVED:** To reduce long error propagation at the time of demodulation and to perform sure data reproduction by rearranging a data stream into which a DSV control bit is inserted so as to make data belonging to the same byte unit be the same modulation unit. **SOLUTION:** A DSV bit decision/insertion part 11 performs DSV control on the data string at a prescribed interval, decides the '1' or '0' of a DSV control bit, inserts it into the data stream at the prescribed interval and outputs it to a data array conversion part 12. Prescribed array conversion is performed there at the prescribed interval of a 'cell' and it is outputted to a modulation part 13 and a SYNC/SYNCID decision part 14. The modulation part 13 modulates the array-converted data string into which the DSV control bit is inserted ad the SYNC/SYNCID decision part 14 decides the pattern of synchronizing signals to be inserted at the prescribed interval. A SYNC bit insertion part 15 inserts the synchronizing signals into modulated signals and an NRZT part 16 NRZI-modulates the signals into which synchronizing signals are inserted and converts them to a recording waveform sequence. COPYRIGHT: (C)2000,JPO

**JAPIO**

© 2005 Japan Patent Information Organization. All rights reserved.  
Dialog® File Number 347 Accession Number 6483269

(19) 日本国特許庁 (J P)

## (12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2000-68847

(P 2 0 0 0 - 6 8 8 4 7 A)

(43) 公開日 平成12年3月3日(2000.3.3)

(51) Int. Cl. <sup>7</sup>

H03M 7/14

識別記号

F I

H03M 7/14

テマコード (参考)

B

審査請求 未請求 請求項の数 9 O L (全15頁)

(21) 出願番号 特願平10-237045

(22) 出願日 平成10年8月24日(1998.8.24)

(71) 出願人 000002185

ソニー株式会社

東京都品川区北品川6丁目7番35号

(72) 発明者 中川 俊之

東京都品川区北品川6丁目7番35号 ソニー株式会社内

(72) 発明者 新福 吉秀

東京都品川区北品川6丁目7番35号 ソニー株式会社内

(74) 代理人 100082131

弁理士 稲本 義雄

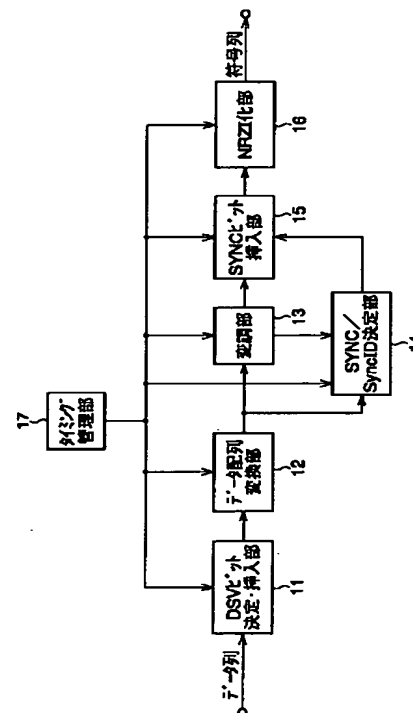
最終頁に続く

(54) 【発明の名称】 変調装置および方法、復調装置および方法、並びに提供媒体

(57) 【要約】

【課題】 復調時の長いエラー伝搬を減らす。

【解決手段】 DSVビット決定・挿入部11は、データ列に1ブロック当たり1ビットのDSV制御ビットを挿入する。データ配列変換部12は、データ列の、同一のバイト単位に属するデータが、同一の変調単位となるようDSV制御ビットが挿入されたデータ列を並び替える。



BEST AVAILABLE COPY

## 【特許請求の範囲】

【請求項 1】 基本データ長が 2 ビットのデータを、最小ランが 1 で、基本符号長が 3 ビットの可変長符号(d, k; m, n; r)に変調する変調装置において、データ列に 1 ブロック当たり 1 ビットの DSV 制御ビットを挿入する DSV 制御ビット挿入手段と、前記データ列の、同一のバイト単位に属するデータが、同一の変調単位となるよう前記 DSV 制御ビットが挿入されたデータ列を並び替える並替手段とを備えることを特徴とする変調装置。

【請求項 2】 前記 DSV 制御ビット挿入手段は、1 ブロックおきに DSV 制御ビットを 2 ビットまとめて挿入することを特徴とする請求項 1 に記載の変調装置。

【請求項 3】 前記並替手段は、1 ブロックおきに、1 ブロックの前記データ列を反転することを特徴とする請求項 1 に記載の変調装置。

【請求項 4】 前記 DSV 制御ビットの値は、変調の後に決定されることを特徴とする請求項 1 に記載の変調装置。

【請求項 5】 基本データ長が 2 ビットのデータを、最小ランが 1 で、基本符号長が 3 ビットの可変長符号(d, k; m, n; r)に変調する変調方法において、データ列に 1 ブロック当たり 1 ビットの DSV 制御ビットを挿入する DSV 制御ビット挿入ステップと、前記データ列の、同一のバイト単位に属するデータが、同一の変調単位となるよう前記 DSV 制御ビットが挿入されたデータ列を並び替える並替ステップとを含むことを特徴とする変調方法。

【請求項 6】 基本データ長が 2 ビットのデータを、最小ランが 1 で、基本符号長が 3 ビットの可変長符号(d, k; m, n; r)に変調する変調装置に、データ列に 1 ブロック当たり 1 ビットの DSV 制御ビットを挿入する DSV 制御ビット挿入ステップと、前記データ列の、同一のバイト単位に属するデータが、同一の変調単位となるよう前記 DSV 制御ビットが挿入されたデータ列を並び替える並替ステップとを含む処理を実行させるコンピュータが読み取り可能なプログラムを提供することを特徴とする提供媒体。

【請求項 7】 最小ランが 1 で、基本符号長が 3 ビットの可変長符号(d, k; m, n; r)を、基本データ長が 2 ビットのデータに復調する復調装置において、変調時のデータ列の並び替えに対応して、復調されたデータ列を並び替える並替手段を備えることを特徴とする復調装置。

【請求項 8】 最小ランが 1 で、基本符号長が 3 ビットの可変長符号(d, k; m, n; r)を、基本データ長が 2 ビットのデータに復調する復調方法において、変調時のデータ列の並び替えに対応して、復調されたデータ列を並び替える並替ステップを含むことを特徴とする復調方法。

【請求項 9】 最小ランが 1 で、基本符号長が 3 ビットの変可変長符号(d, k; m, n; r)を、基本データ長が 2 ビットのデータに復調する復調装置に、変調時のデータ列の並び替えに対応して、復調されたデータ列を並び替える並替ステップを含む処理を実行させるコンピュータが読み取り可能なプログラムを提供することを特徴とする提供媒体。

## 【発明の詳細な説明】

## 【0 0 0 1】

10 【発明の属する技術分野】本発明は、変調装置および方法、復調装置および方法、並びに提供媒体に関し、特に、データ伝送や記録媒体への記録に適するようにデータを変調し、再生する変調装置および方法、復調装置および方法、並びに提供媒体に関する。

## 【0 0 0 2】

【従来の技術】データを所定の伝送路で伝送する、または、例えば磁気ディスク、光ディスク、光磁気ディスク等の記録媒体に記録するとき、伝送や記録に適するようにデータの変調が行われる。このような変調方法の 1 つとして、ブロック符号が知られている。このブロック符号は、データ列を  $m \times i$  ビットからなる単位（以下、単にデータ語と称する）にブロック化し、このデータ語を適当な符号則に従って  $n \times i$  ビットからなる符号語に変換するものである。そして、この符号は、 $i$  が 1 のとき、固定長符号となり、また、 $i$  が複数個選べるとき、すなわち、1 乃至  $i_{\max}$ （最大の  $i$ ）の範囲の所定の  $i$  を選択して変換したとき、可変長符号となる。このブロック符号化された符号は、可変長符号(d, k; m, n; r)と表される。

30 【0 0 0 3】ここで  $i$  は、拘束長と称され、 $i_{\max}$  は、最大拘束長  $r$  と称される。また、最小ラン  $d$  は、符号系列内の連続する“1”の間に入る、連続する“0”の最小の個数を示し、最大ラン  $k$  は、符号系列内の連続する“1”の間に入る、連続する“0”の最大の個数を示している。

【0 0 0 4】コンパクトディスクやミニディスク等においては、上述のようにして得られた可変長符号に対して、“1”で反転、“0”で無反転とする NRZI (Non Return to Zero Inverted) 変調を行い、NRZI 変調された可変長符号（以下、NRZI 変調された可変長符号を、レベル符号と称する）を記録する。

【0 0 0 5】また、レベル符号を、“1”から“0”あるいは“0”から“1”に反転したとき、即ち、エッジとなったとき、“1”とする、逆 NRZI 変調を行うと、元の EFM 符号や RLL (1-7) 符号と同じ符号列を得ることができる。この逆 NRZI 符号列は、エッジ符号と称する。

50 【0 0 0 6】レベル符号の最小反転間隔を  $T_{\min}$  とし、最大反転間隔を  $T_{\max}$  とするとき、線速方向に高密度の記録を行うためには、最小反転間隔  $T_{\min}$  は長い方が、すなわち最小ラン  $d$  は大きい方が良く、また、クロックの再生の面からは最大反転間隔  $T_{\max}$  は短いほうが、すなわち最

大ランクは小さい方が望ましく、種々の変調方法が提案されている。

【0007】例えば、磁気ディスクまたは光磁気ディスク等の記録で用いられる変調方式としてRLL(1-7)がある。この変調方式のパラメータは(1,7;2,3;2)であり、 $(d+1)T$ で求められる最小反転間隔 $T_{min}$ は、 $(1+1)T$ より、 $2T$ となる。データ列のビット間隔を $T_{data}$ とすると、この最小反転間隔 $T_{min}$ は、 $(m/n) \times T_{min} = (2/3) \times 2$ より、 $1.33T_{data}$ となる。また、 $(k+1)T$ で求められる最大反転間隔 $T_{max}$ は $8(=7+1)T(=2/3 \times 8T_{data} = 5.33T_{data})$ となる。さらに、 $(m/n)T$ で求められる検出窓幅 $T_w$ は、 $0.67(=2/3)T_{data}$ となる。

【0008】RLL(1-7)符号の変換テーブルは、例えば、表1に示すようなテーブルである。

【0009】

<表1>

RLL(1,7; 2,3; 2)

	データ	符号
i=1	11	00x
	10	010
	01	10x
i=2	0011	000 00x
	0010	000 010
	0001	100 00x
	0000	100 010

ここで変換テーブル内の記号xは、次に続くチャンネルビットが"0"であるとき"1"を与え、または次に続くチャンネルビットが"1"であるとき"0"を与える。最大拘束長rは2である。

【0010】ところで、RLL(1-7)による変調を行ったチャンネルビット列は、発生頻度としては $T_{min}$ である $2T$ が一番多く、以下 $3T$ 、 $4T$ と続く。 $2T$ や $3T$ のようなエッジ情報が早い周期で多く発生すると、クロック再生には有利であるが、 $2T$ が連続しつづけると、記録波形に歪みが生じやすくなる( $2T$ の波形出力は小さく、デフォーカスやタンジェンシャル・チルトによる影響を受けやすい)。また、さらに高線密度で、最小マークの連続した記録は、ノイズ等の外乱の影響を受けやすく、データ再生誤りを起こしやすくなる。

【0011】そこで、本出願人は特願平9-133379号として、 $T_{min}$ が所定の回数以上連続するのを制限することを提案したが、その符号であるRML(1-7)の変換テーブルは、例えば、表2に示すテーブルである。

【0012】

<表2>

RML(1,7;2,3;3)

	データ	符号
i=1	11	00x
	10	010
	01	10x
i=2	0011	000 00x
	0010	000 010
	0001	100 00x
	0000	100 010
i=3	100110	100 000 010

ここで変換テーブル内の記号xは、次に来るチャンネルビットが"0"であるとき"1"を与え、また次に来るチャンネルビットが"1"であるとき"0"を与える。最大拘束長rは3である。

【0013】表2を使用した変換は、データ列が"100"となった場合、さらに次の4データを参照し、合計6データ列が"100110"となったとき、最小ランクの繰返しを制限するコード"100 000 010"を与える。この変換により得られる符号の最小ランクの繰返しは、最大で5回までになる。

【0014】ところで、記録媒体への記録およびデータの伝送の際には、各媒体(伝送)に適した符号化変調が行われるが、これら変調符号に直流成分が含まれているとき、たとえばディスク装置のサーボの制御におけるトラッキングエラーなどの、各種のエラー信号に変動が生じやすくなったり、あるいはジッターが発生しやすくなったりする。従って、直流成分はなるべく含まないほうが良い。

【0015】ここで上記に示した、可変長の最小ランク=1で、変換率 $m=2$ 、および $n=3$ のRLL符号は、DSV(Digital Sum Value)制御が行われていない。DSV制御とは、チャンネルビット列をNRZI化し(すなわちレベル符号化する)、そのビット列(データのシンボル)の"1"を+1、"0"を-1として符号を加算していったときその総和(DSV)の絶対値を小さくする制御を意味する。DSVは符号列の直流成分の目安となり、DSVの絶対値を小さくすることは、符号列の直流成分を抑制することとなる。

【0016】このDSV制御を行うDSV制御ビットは、通常、 $2 \times (d+1)$ ビット使用され、例えば、 $d=1$ の場合、 $2 \times (1+1) = 4$ ビットである。このとき、任意の間隔において、最小ランおよび最大ランを守ることができ、かつ反転または非反転も可能な完全なDSV制御が行われる。

【0017】しかし、DSV制御ビットは、基本的には冗長ビットである。従って符号変換の効率から考えれば、DSV制御ビットはなるべく少ないほうが良い。

【0018】そこで、DSV制御ビットを、 $1 \times (d+1)$ 、すなわち $d=1$ の場合では、 $1 \times (1+1) = 2$

ビットとしても、任意の間隔において、反転／非反転も可能な完全なDSV制御が行われる。ただし、最小ランは守られるが、最大ランは大きくなり、 $(k+2)$ となる。記録符号として最小ランは必ず守る必要があるが、最大ランについてはその限りではない。場合により、最大ランを破るパターンを同期信号に用いるフォーマットが存在する（DVDのEFMプラスは最大ラン11Tだが、フォーマットの都合上14Tを許している）。

【0019】そして、表2のRML符号の基本性能を保ったまま、これらよりもさらに効率が良くDSV制御を行えるテーブルとして、17PP (Parity Preserve) 符号がある。17PP符号は、ラン制限 $d=1$ 、 $k=7$ であり、その上に最小ランの連続を制限し、さらにデータ語と符号語の対応した要素に規則を与えた変調符号である。

【0020】本出願人が特願平10-150280号にて提案している17PP符号の変換テーブルは、例えば、以下の通りである。

【0021】<表3>

17PP-32(1,7;2,3;4)

データ	符号
11	*0*
10	001
01	010
0011	010 100
0010	010 000
0001	000 100
000011	000 100 100
000010	000 100 000
000001	010 100 100
000000	010 100 000
"110111	001 000 000(next010)
00001000	000 100 100 100
00000000	010 100 100 100
if xxl then *0* = 000	
xx0 then *0* = 101	

-----  
 "110111 001 000 000(next010):  
 When next channel bits are '010',  
 convert '11 01 11' to '001 000 000' after  
 using main table and termination table.  
 -----

【0022】表3は、最小ラン $d=1$ 、最大ラン $k=7$ で、変換テーブル内の要素に不確定符号を有する。不確定符号は、変換するデータ列2ビットが(11)であったとき、その直前の符号語列によって"000"あるいは"101"が選択される。直前の符号語列の1チャンネルビットが"1"であったとき、最小ランを守るために、(11)の変換は、"000"となる。また直前の符号語列の1チャンネルビットが"0"であったとき

は、"101"とし、最大ランを守れるようにする。

【0023】表3の変換テーブルは、可変長構造のテーブルである。すなわち拘束長 $i=1$ における変換コードは、必要数の4つ $(2^{\lceil \log_2 m \rceil} \times i) = 2^{\lceil \log_2 2 \rceil} \times 1 = 4$ よりも少ない3つで構成されている。すなわちデータ列を変換する際に、拘束長 $i=1$ だけでは変換出来ないデータ列が存在する。結局、表3の変換テーブルにおいて、全てのデータ列に対応するため、すなわち変換テーブルとして成り立つためには、拘束長 $i=3$ までを要する。

【0024】また、表3の変換テーブルは、変換テーブル内に、最小ランの連続を制限する、置き換えコードを有する。例えば、データ列(110111)は、さらに後ろに続く符号語列を参照し、それが"010"であったとき、"001 000 000"に置き換えられる。後ろに続く符号語列が"010"以外であれば"\*0\* 010 \*0\*"に変換される。これによって、データ変換後の符号語列は、最小ランの連続が制限され、最小ランの繰り返しは、最大でも6回までとなる。

【0025】さらに表3の変換テーブルは、データ列の要素内の"1"の個数と、変換される符号語列の要素内の"1"の個数が、それを2で割った時の余りが、どちらも1あるいは0で同一となるような変換規則を有する。例えば、データ列の要素(000001)は"010 100 100"の符号語列に対応しているが、それぞれ"1"の個数は、データ列では1個、対応する符号語列では3個であり、どちらも2で割った余りが1で一致する。同様に、データ列の要素(000000)は、"010 100 000"の符号語列に対応しているが、それぞれ"1"の個数は、データ列が0個、対応する符号語列は2個であり、どちらも2で割った余りが0で一致する。

【0026】そして、表3の変換テーブルは、最大拘束長 $r=4$ である。 $i=4$ の変換コードは、最大ラン $k=7$ を実現するための、置き換えコードを有する。

【0027】表3の変換テーブルに従ってデータ列を変調し、変調後のチャンネルビット列を、所定の間隔で、これまでと同様にDSV制御することができるが、データ列と、変換される符号語列の関係を生かして、さらに効率良くDSV制御を行うことができる。

【0028】すなわち、変換テーブルが、データ列の要素内の"1"の個数と、変換される符号語列の要素内の"1"の個数が、それを2で割った時の余りが、どちらも1あるいは0で同一となるような変換規則を有するとき、チャンネルビットで、「反転」を表す"1"、あるいは「非反転」を表す"0"のDSV制御ビットを挿入することは、データビット列内に、「反転」するならば"1"を挟み、「非反転」ならば"0"のDSV制御ビットを挿入することと等価になる。

【0029】たとえば表3において、データ変換する3ビットが"001"と続いたとき、その後ろにおいてDS

V制御ビットを挟むとすると、データ変換は、 $(001-x)$  ( $x$ は1ビットで、0または1)となる。ここで  $x$ に"0"を与えれば、表3の変換テーブルは、

データ	符号
0010	010 000

となり、また、"1"を与えれば、

データ	符号
0011	010 100

となる。符号語列をNRZI化してレベル符号化したとき、これらは

データ	符号	レベル符号
0010	010 000	011111
0011	010 100	011000

となり、レベル符号列の最後の3ビットが相互に反転している。すなわち、DSV制御ビット  $x$ の、"1"と"0"を選択することによって、データ列内においても、DSV制御を行うことができる。

【0030】DSV制御による冗長度を考えると、データ列内において1ビットのDSV制御を行うことは、チャンネルビット列で表現すれば、表3では変換率  $m=2$ 、 $n=3$ であるから、1.5チャンネルビットでDSV制御を行うことに相当する。ここで例えば表1のようなRLL(1-7)テーブルにおいてDSV制御をおこなうためには、チャンネルビット列においてDSV制御を行うことになり、最小ランを守るためには、前述の通り、少なくとも2チャンネルビットが必要であり、冗長度は、より大きくなる。

【0031】表3の変換テーブルは、データ列内でDSV制御が行えるので、効率の良いDSV制御が行えると共に、最小ランの繰返しが制限されているので、高線密度記録再生に適している符号を生成する。

【0032】今、上記の方法でDSV制御を行うとき、データ列内においてDSV制御ビットの1ビットが挿入される。ここで表3のような17PP符号は変換率は  $m=2$ 、 $n=3$ であり、変換の基本単位は2ビットである。すなわち、表3にもとづくデータ変換は、2ビット単位でデータを進めて行われる。このときデータ列内にDSV制御ビットとして1ビットが入ったとき、本来2ビット単位であった部分が1つずれてしまう。これはデータ変換時には影響がないが、復調時にビットシフトによるエラーが発生すると、これを復調したとき、長いエラー伝搬を起こしやすくなる。

【0033】

【発明が解決しようとする課題】以上のように、磁気ディスクや光磁気ディスク、光ディスク等の記録媒体を高密度化し、変調符号として、記録・再生時の歪みを少なくすることでエラーの発生を抑え、より高密度記録再生に適した符号である、17PP符号を選択した場合、DSV制御の方法として Parity Preserveを生かして効率の良い制御を行うとき、データ列内に挿入される1ビットは、復調時において、長いバイトエラー伝搬を起こしや

すくなるという欠点が生じる。

【0034】本発明はこのような状況に鑑みてなされたものであり、復調時の長いエラー伝搬を減らし、より確実なデータの再生ができるようにすることを目的とする。

【0035】

【課題を解決するための手段】請求項1に記載の変調装置は、データ列に1ブロック当たり1ビットのDSV制御ビットを挿入するDSV制御ビット挿入手段と、データ列の、同一のバイト単位に属するデータが、同一の変調単位となるようDSV制御ビットが挿入されたデータ列を並び替える並替手段とを備えることを特徴とする。

【0036】請求項5に記載の変調方法は、データ列に1ブロック当たり1ビットのDSV制御ビットを挿入するDSV制御ビット挿入ステップと、データ列の、同一のバイト単位に属するデータが、同一の変調単位となるようDSV制御ビットが挿入されたデータ列を並び替える並替ステップとを含むことを特徴とする。

【0037】請求項6に記載の提供媒体は、変調装置に、データ列に1ブロック当たり1ビットのDSV制御ビットを挿入するDSV制御ビット挿入ステップと、データ列の、同一のバイト単位に属するデータが、同一の変調単位となるようDSV制御ビットが挿入されたデータ列を並び替える並替ステップとを含む処理を実行させるコンピュータが読み取り可能なプログラムを提供することを特徴とする。

【0038】請求項7に記載の復調装置は、変調時のデータ列の並び替えに対応して、復調されたデータ列を並び替える並替手段を備えることを特徴とする。

【0039】請求項8に記載の復調方法は、変調時のデータ列の並び替えに対応して、復調されたデータ列を並び替える並替ステップを含むことを特徴とする。

【0040】請求項9に記載の提供媒体は、復調装置に、変調時のデータ列の並び替えに対応して、復調されたデータ列を並び替える並替ステップを含む処理を実行させるコンピュータが読み取り可能なプログラムを提供することを特徴とする。

【0041】請求項1に記載の変調装置、請求項5に記載の変調方法、および請求項6に記載の提供媒体においては、データ列に1ブロック当たり1ビットのDSV制御ビットを挿入し、データ列の、同一のバイト単位に属するデータが、同一の変調単位となるようDSV制御ビットが挿入されたデータ列を並び替える。

【0042】請求項7に記載の復調装置、請求項8に記載の復調方法、および請求項9に記載の提供媒体においては、変調時のデータ列の並び替えに対応して、復調されたデータ列を並び替える。

【0043】

【発明の実施の形態】以下に本発明の実施の形態を説明するが、特許請求の範囲に記載の発明の各手段と以下の

実施の形態との対応関係を明らかにするために、各手段の後の括弧内に、対応する実施の形態（但し一例）を付加して本発明の特徴を記述すると、次のようになる。但し勿論この記載は、各手段を記載したものに限定することを意味するものではない。

【0044】すなわち、請求項1に記載の変調装置は、データ列に1ブロック当たり1ビットのDSV制御ビットを挿入するDSV制御ビット挿入手段（例えば、図4のDSVビット決定・挿入部11）と、データ列の、同一のバイト単位に属するデータが、同一の変調単位となるようDSV制御ビットが挿入されたデータ列を並び替える並替手段（例えば、図4のデータ配列変換部12）とを備えることを特徴とする。

【0045】請求項7に記載の復調装置は、変調時のデータ列の並び替えに対応して、復調されたデータ列を並び替える並替手段（例えば、図6のデータ配列変換部35）を備えることを特徴とする。

【0046】図1は、例えば、表3の変換テーブルを用いて変調したときの、記録符号列フォーマットの一例を表す図である。図1の処理では、17PP方式の特性を利用して、データ列内に1ビットのDSV制御ビット(DC-bit)を挿入することでDSV制御を行っている。また、データ変換した後、同期信号とIDを含めたチャンネルビット列に、一定の間隔でDSV制御が行われるようになっている。DSV制御の間隔は、データ語で45データである。これは符号語で言えば、69符号語  $((45+1) \times 1.5=69)$  間隔になる。69符号語となるようにデータを挟むと、これに合わせて同期信号部分のデータ列は、25データ  $(69 = (20 + 25) \times 1.5)$  となる。そして45データを27cell、これに（同期信号+25データ）の1cellとを合わせて、1Sync当たり、1240データで、1つのフレームを構成している。同期信号は、例えば、後述の表4にあるパターンとし、最初の2ビットが接続用のビットで、続く2T-9T-9Tが同期信号検出のためのパターンとなる。そしてこれら24ビットに続く6ビットは、複数種類の同期信号を持った場合のSync-ID用のビットとして与えられている。

【0047】表3における、複数種類の同期信号パターンの例を示す。決定する同期信号パターンは、最小ランが守られるとともに、最小ランの繰返しが表3の変換テーブルにあるように、6回までに制限されるように選択する。また、最大ランは、同期信号検出パターン以外では発生しないように選択する。

【0048】各同期信号パターン同士の距離が2以上になるように同期信号パターンを選択したとき、変換テーブルは、例えば、表4のようになる。ここで、距離が2以上とは、それぞれの同期信号パターンにおいて、検出されたとき（再生データはレベル符号）に、同期信号の30チャンネルビット中で少なくとも2ヶ所以上が異なっていることを言う。表4の場合、同期信号パターンは、

後ろ6ビットでこのような条件を満たすようにパターンを選択する。

【0049】

＜表4＞

17PP.RML.32

30channel-bit Syncs

```
#01 010 000 000 010 000 000 010 000 001
                                000 100
                                001 001
                                010 000
                                010 010
                                100 001
                                101 000
```

# = 0 not terminate case

# = 1 terminate case

Termination table

```
00    000
0000 010 100
```

【0050】表4の同期信号パターンの先頭の24ビットである同期信号検出部分”#0101000000001000000010”について説明する。先頭の”#”は接続用のビットで、0か1のどちらかを与える。2チャンネルビット目は、最小ランを守るために”0”を与える。3チャンネルビット目と4チャンネルビット目で2Tを与える。そして5チャンネルビット目から、同期信号パターンとして、 $k=8$ となる9Tを2回連続して与える。すなわち”1”と”1”の間に、”0”が8つ並ぶ。これを2回続ける。同期信号パターンの最後のチャンネルビットの”1”は、最大ランを決定する。ここまでで23チャンネルビットである。さらに、最後に1ビット”0”を付加する。これによって、以降のビットに関わらず、最小ラン $d=1$ を守ることができる。

【0051】ここで終端用テーブルと、同期信号パターンの接続用ビット”#”の説明をする。終端用テーブルは、表4にあるように、

```
00    000
0000 010 100
```

となる。終端用テーブルが必要となるのは、最小ランの連続を制限するなどのための置き換えコードでない変換コードの存在する拘束長 $r$ のそれぞれにおいて、変換コードが4つよりも小さいようなときである。すなわち表3では、拘束長 $i=1$ における変換コードは3つであるから終端用テーブルが必要となる。また拘束長 $i=2$ における変換コードも3つであるから終端用テーブルが必要となる。拘束長 $i=3$ における変換コードは5つあり、そのうち1つが置き換えコードで、4つが変換コードであり、必要数を持っているので終端される。拘束長

$i = 4$ における変換コードはいずれも置き換えコードであるため、終端を考慮しなくてよい。従って、終端用テーブルには、拘束長  $i = 1$  の (0 0) と  $i = 2$  の (0 0 0 0) を与える。

【0 0 5 2】同期信号パターンの接続用ビット”#”は、終端用のテーブルを用いる場合と用いない場合を区別するために与える。すなわち同期信号として与えられた、先頭の1チャンネルビット目の”#”は、終端コードを用いたときは「1」を与え、そうでないときは「0」を与える。こうすることによって、復調時において、間違

いなく終端用のテーブルを用いる場合と用いない場合を識別することができる。

【0 0 5 3】そして、同期信号として複数種類が必要な場合、(2 3 + 1) チャンネルビットに加えて、さらに6ビットを追加し、3 0 チャンネルビットを与える。6ビットのID部分は、距離が2以上とれるように選ばれ、7種類が得られる。表4は、以上のように構成されている。

【0 0 5 4】ところで、表3の変換テーブルにより生成される1 7 P P (Parity Preserve) 符号は、変換率  $m = 2$ 、 $n = 3$  であることより、変換の最小実行単位は、2ビットである。これはさらに、バイト単位でみた場合は、8ビットが実行単位である。

【0 0 5 5】図2は、ビット配列の変換例を示した図である。DSV制御単位を(1 9 + 1) ビットとし、これを「# 0、cell」、および「# 1 cell」などと表す。またデータ列をバイト単位ごとにまとめたものを「0」、「1」、および「2」などと表し、「0」の対応ビットを「00, 01, 02, ..., 07」、「1」の対応ビットを「10, 11, ..., 17」、および「2」の対応ビットを「20, 21, ..., 26, 27」と表す。またDSV制御ビットを「d0」、「d1」と表し、その他の「ID」は同期信号ID用ビットを表している。

【0 0 5 6】図2 (A) は、図1の場合のビット配列を示す図であり、1 9 データごとにDSV制御ビットが1ビットが挟まれている。このとき、「# 1 cell」は、例えば2バイト目最後と3バイト目先頭の境目である、「27-30」という2ビット単位の組合せを含む。ここで、仮に復調時にシフトエラーが発生し、またその結果、2ビット単位の復調エラーが発生したとする。図2 (A) の「# 0 cell」の各2ビット単位のように、通常1ヶ所の2ビット単位エラーは、1バイトのエラーとなるが、図2 (A) 「# 1 cell」のバイトの境目「27-30」で発生したケースでは、1ヶ所の2ビット単位エラーが、2バイト連続のエラーとなる。図2 (A) は、また同様に、「# 1 cell」内の3バイト目と4バイト目の境目である「37-40」でも、1ヶ所の2ビット単位エラーが、2バイト連続のエラーとなる。このように、最小の2ビット単位の中に、隣り合うバイトの境目が存在する場合、この2ビット単位で発生したエラーは、長い

バイトエラーの伝搬を発生させる。

【0 0 5 7】そして、図2 (A) のような、1ヶ所の2ビット単位エラーから長いバイトエラーを引き起こすことの無いようにしたのが、図2 (B)、図2 (C)、および図2 (D) に示すビット配列である。

【0 0 5 8】図2 (B) に示すビット配列は、「# 0 cell」と「# 1 cell」の2つを単位とし、「# 0 cell」の最後にDSV制御ビットを挟まず、代わりにそこに置かれる20番目のデータ「23」を反転する／しないによってDSV制御を行うものである。そして「# 1 cell」の最後の2ビットは、DSV制御ビットとして与えられ、先ほどの「23」の反転する／しないの情報が「d0」に、「# 1 cell」のDSV制御結果が、「d1」に与えられる。図2 (B) のような形式にしたとき、2ビット単位でのずれは発生をしないので(2バイトにまたがる変換の最小実行単位は存在しないので)、1ヶ所の2ビット単位エラーが2バイトのエラーとなる恐れがない。

【0 0 5 9】図2 (B) を、図1にあるフォーマットのように書き直したものが、図3である。すなわち、1 Sync を1 2 4 0 データとした場合である。

【0 0 6 0】この他に、図2 (C) に示すデータ配列は、「# 1 cell」、「# 3 cell」、および「# 5 cell」、などと、1つおきの「cell」において、データの列を最後から逆に配列したものである。すなわち、そのままではビットずれが含まれてしまう「cell」に対してのみ、逆に配列されている。これは実質的に、「# 0 cell」と「# 1 cell」の2つのセルが1単位の構成となる。図2 (C) に示すように、「# 1 cell」は、先頭が4バイト目に相当の「45-44」の組合せから始まり、「43-42」、「41-40」と続き、「# 1 cell」の最後は、2バイト目相当の「25-24」の組合せに続いて、「23-d1」の組合せで終了する。DSV制御ビット「d0」、「d1」の位置は、図2 (A) のデータ配列と同様である。図2 (C) のような形式にしたとき、2ビット単位でのずれは発生をしないので、1ヶ所の2ビット単位エラーが2バイトのエラーとなる恐れがなくなる。

【0 0 6 1】さらに、図2 (D) に示すデータ配列は、各「cell」の最後にDSV制御の1ビットと、さらに同期信号ID用の1ビットを与え、2ビットを各「cell」に与えることで、2ビット単位を守るようにしたものである。図2 (D) のような形式にしたとき、2ビット単位でのずれは発生をしないので、1ヶ所の2ビット単位エラーが2バイトのエラーとなる恐れがなくなる。

【0 0 6 2】実際のビットシフト等による復調エラー伝搬は、2ビット単位だけではなく、数ビット単位に渡る場合がある。図2 (A) に示すデータ配列では、復調のとき、2ビットのエラーのみであっても、バイトエラーの増加を招く箇所がある。例えば、図2 (A) の「# 1 cell」の「27-30」および「37-40」で発生したエラーは、それぞれ、2バイト連続エラーを引き起こす。ま



た、2ビットより多い復調エラー伝搬では、同様に、「#1 cell」の「27-30」および「37-40」において、より3バイト連続エラーを引き起こしやすくする。

【0063】これに対し、図2(B)乃至(D)では、データフォーマットにおける、2ビット単位を守るようにしたため、上のような連続エラーを引き起こさないようにすることが出来る。しかし、その一方では、それぞれ次のような特性を持つことになる。

【0064】図2(B)は、「#1 cell」の最後の、「d0」付近で長い復調エラー伝搬が発生したときに、「#0 cell」の最後のデータ「23」に伝搬するので、2バイト目と4バイト目に分散してエラーが発生する可能性を有する。

【0065】図2(C)は、「#0 cell」の最後の、「d0」付近で長い復調エラー伝搬が発生したときに、「#0 cell」の最後付近のデータの前後「22」や「45」の、2バイト目と4バイト目に分散してエラーが発生する可能性を有する。

【0066】図2(D)は、各「cell」で単独なので、上のような、長い復調エラー伝搬により複数バイトへエラーの影響は及ぼさないが、各「cell」に2ビットの冗長を入れることより、「cell」の数が多いほど冗長になる。例えば、表4の変換テーブルでは、6ビットの同期信号IDが与えられるので、1つのSync当たり、「cell」の数が6個よりも大きいと、図2(D)のデータ配列は、図2の他の方式に較べて、さらに冗長となる。図1に示したデータ配列は、その冗長となる一例となる。また、ID情報が各「cell」の最後に分散されるため、同期信号識別箇所が増加する。

【0067】これらについては、後述するシミュレーションにて検討を行っている。

【0068】本発明に係る変調装置の一実施の形態を図面を参照しながら説明する。この実施の形態は、データ列を表3の変長符号(d, k; m, n; r) = (1, 7; 2, 3; 4)に変換する変調装置に本発明を適用したものである。

【0069】図4は、この変調装置の構成を示すブロック図である。DSVビット決定・挿入部11は、データ列より、所定の間隔でDSV制御を行い、DSV制御ビットの“1”あるいは“0”を決定し、データ列に所定の間隔で挿入し、データ配列変換部12に出力する。データ配列変換部12は、DSV制御されたデータ列を所定の「cell」の間隔で、所定の配列変換を行い、変調部13およびSYNC/SyncID決定部14に出力する。変調部13は、DSV制御ビットが挿入され、配列変換されたデータ列を変調する。SYNC/SyncID決定部14は、所定の間隔で挿入される同期信号(Sync)のパターンを決定する。SYNCビット挿入部15は、変調された信号に同期信号を挿入する。NRZI化部16は、同期信号が挿入された信号をNRZI変調し、記録波形列に変換する。タイミング管理部1

7は、タイミング信号を生成し、DSVビット決定・挿入部11、データ配列変換部12、変調部13、SYNC/SyncID決定部14、SYNCビット挿入部15、およびNRZI化部16に供給し、タイミングを管理する。

【0070】図4におけるデータ配列変換部12は、図2(B)乃至図2(D)に示した形式でデータ配列の変換を行う。図4におけるSYNC/SyncID決定部14は、表4に示した30ビットの同期信号パターンから、挿入する同期信号のパターンを決定する。SYNCビット挿入部15は、上に説明したようにして決定された、SYNC/SyncID決定部14から供給された同期信号を挿入する。同期信号が挿入された後の次の変換テーブルは、先頭からスタートするようになっており、すなわち同期信号の挿入により変調は終端される。

【0071】次に、この変調装置の動作について説明する。

【0072】データ列は、所定の間隔でDSV制御が行われ、さらにまた、所定の間隔で同期信号が挿入される。DSVビット決定・挿入部11は、ある位置までの積算DSVと、次の所定の間隔の区間DSVを計算し、これらを合わせたDSV値が小さくなる方のDSV制御ビットの“1”あるいは“0”を決定し、これをデータ列に挿入する。DSV値は、データ列だけでは判定できないので、データ列より変換テーブルを用いて符号語列を発生させ、これよりDSV値を求める。なおデータ列は、本発明にあるようなデータ配列変換を行った後の、確定したデータの並びを元にDSV計算が行われている。

【0073】例えば図2(B)のような場合、データ配列変換の処理は、「#0 cell」の最後のDSV制御で、次の「#1 cell」の区間DSVが計算され、区間DSVおよび積算DSV値により、図2(B)中の「23」のデータを、DSV制御ビットの代わりとして反転させるか否かが決定され、その情報を「#1 cell」の後方にある「d0」に挿入する。例えば反転したとき「d0」に“1”が設定され、反転しないとき「d0」に“0”が設定される。あらかじめ決定しておけば、その逆でもかまわない。それ以外でもかまわない。一方、「#1 cell」の最後にDSV制御ビットを入れるとき、次の「#2 cell」の区間DSVが計算され、また積算DSV値より、図2(B)中の「d1」においてDSV制御ビットが設定される。もちろんこのとき、「d0」のビットは決定された後であるから、値に変動はない。

【0074】例えば図2(C)のような場合、データ配列変換の処理は、「#0 cell」の最後にDSV制御ビットを入れるとき、次の「#1 cell」のデータ列を先頭から最後まで逆に入れ替え、その後区間DSVを計算し、また積算DSV値より、図2(C)中の「d0」のDSV制御ビットを決定する。このときの「#1 cell」の区間DSVは、図2(C)に示したように、データ配列を逆に変換したものを計算し、計算される。一方、「#1 cell」の最

後へのDSV制御ビットの挿入では、次の「# 2 cell」のデータ列の変換（入れ替え）は行わず、区間DSVが計算される。そして積算DSV値より、図2（C）中の「d1」のDSV制御ビットが決定される。

【0075】例えば図2（D）のような場合、データ配列変換は、各「cell」の最後に、IDビットとDSV制御ビットを挟む。IDビットは、その「cell」の先頭同期信号が複数種類あるときの、識別用の情報ビットであり、各「cell」の最後においてDSV制御ビットを挟むように、その前に配置される。この方式の場合は、SYNCビット挿入部15は、同期信号（24ビット）のみを挿入し、またSyncIDの決定は、実質的には、データ配列変換部12において行われることになる。

【0076】DSV値の挿入、およびデータ配列変換が行われたビット列は、次の変調部13において、変換テーブルによって変調され、SYNCビット挿入部15に供給される。また変調部13は、同期信号の間隔を記憶し、同期信号付近まで変調を行うが、通常の変換テーブルで変換できない場合、すなわち表4の終端テーブルを用いる必要がある場合、その情報をSYNC/SyncID決定部14に供給する。

【0077】SYNC/SyncID決定部14は、同様に同期信号の間隔を記憶し、同期信号の挿入される直前の状態によって、同期信号の先頭の接続ビットを決定する。通常の変換テーブルでデータ変換を行った場合、先頭の接続ビットには”0”が設定される。通常の変換テーブルで行うことができず、終端テーブルを用いる必要があるとき、内蔵する終端テーブルが参照され、データ列は、変換される。このとき、先頭の接続ビットには”1”が設定される。

【0078】このように同期信号のうち前の24ビットまでが決定される。そしてさらに、後の6ビットは、複数種類の同期信号のSyncIDビットとして決定される。SyncIDビットには、例えば、表4のように、それぞれが互いに距離2を持った、7種類の同期信号パターンが与えられる。

【0079】このようにして同期信号は決定され、以上のように決定された同期信号ビットは、SYNCビット挿入部15において、変調された信号に挿入される。SYNC/SyncID決定部14に内蔵された終端テーブルを用いて同期信号が決定された場合、SYNCビット挿入部15は、終端テーブルの参照を反映した同期信号を挿入すれば良い。

【0080】最後に、NRZI化部16は、これらDSV制御が行われ、さらに同期信号が挿入されたチャンネルビット列を記録符号に変換する。

【0081】図5は、変調装置の他の実施の形態の構成を示すブロック図である。図4を用いて説明したように、DSV制御ビットの決定は、DSV値計算のために、変調およびNRZI化を行う必要がある。さらに同期信号も、DS

V制御されるので、やはりNRZI化される必要がある。従って、変調装置は、図5のように構成できる。

【0082】コントロールビット挿入部21は、所定のビット数単位でDSV制御を行うビットを与えて、データ配列変換部12へ出力する。このビット数単位は、Syncビットをも含んで考慮されるので、必ずしも種類限りのビット数でなくてもよい。データ配列変換部12は、所定のデータ配列変換を行い、これを変調部13へ出力する。変調部13は、データ配列変換部12より得られた、データ列をデータ変換し、チャンネルビット列を作成する。変調部13において、Syncの直前においてデータ変換が出来なかったとき、終端テーブルが用いられる。

【0083】SYNC/SyncID挿入部22は、同期信号を、変調された符号語の所定の間隔で挿入する。SYNC/SyncID挿入部22はまた、終端テーブルを有し、必要に応じて終端テーブルを用いて変調し、30ビットの同期信号パターンを挿入する。同期信号、およびDSV制御ビットを含んだ符号語列は、NRZI化部16でレベル符号化される。そしてDSVビットSYNC決定部23は、送られてきたレベル符号化列をもとにDSV値を計算し、最終的にDSV値を決定する。DSVビットSYNC決定部23の出力値は、記録符号列であり、図4の変調装置の最終出力値と同じ結果である。タイミング管理部17は、タイミング信号を生成し、コントロールビット挿入部21、データ配列変換部12、変調部13、SYNC/SyncID挿入部22、NRZI化部16、およびDSVビットSYNC決定部23に供給し、タイミングを管理する。

【0084】次に、その動作を説明する。コントロールビット挿入部21は、入力されたデータ列より、所定の間隔で挿入されるDSV制御ビットに”1”を設定したビット列、およびDSV制御ビットに”0”を設定したビット列を作成する。この2種類のデータ列は、次のデータ配列変換部12で、例えば図2（B）乃至図2（D）の形式のように、配列が変換される。

【0085】例えば図2（B）のような場合、データ配列変換部12の配列変換は、「#0cell」の最後のDSV制御で、次の「#1cell」の区間DSVが計算され、区間DSVおよび積算DSV値によって、図2（B）中の「23」のデータを、DSV制御ビットの代わりとして反転させるか否かが決定され、その情報を「#1cell」の後方にある「d0」に挿入する。例えば反転したとき「d0」に”1”が設定され、反転しないとき「d0」に”0”が設定される。あらかじめ決定しておけば、その逆でもかまわない。一方、「#1cell」の最後にDSV制御ビットを入れるとき、次の「#2cell」の区間DSVが計算され、また積算DSV値より、図2（B）中の「d1」においてDSV制御ビットが設定される。もちろんこのとき「d0」のビットは決定された後であるから、値に変動はない。そして図2（B）の形式では、コントロールビット挿入部21により、挿入される”1”又は”0”は、図2（B）の

「# 0 cell」の「23」では挿入されず、「23」データ自身の(1)と(0)の2種類のデータ列が作成される。また「# 1 cell」の最後のビットは、通常のDSV制御ビット挿入の方法で挿入される。

【0086】例えば図2(C)のような場合、「# 0 cell」の最後にDSV制御ビットを入れるとき、次の「# 1 cell」のデータ列を先頭から最後まで逆に入れ替え、その後区間DSVを計算し、また積算DSV値より、図2(C)中の「d0」のDSV制御ビットを決定する。このときの「# 1 cell」の区間DSVは、図2(C)に示したように、データ配列を逆に変換したものを変調し、計算される。一方、「# 1 cell」の最後へのDSV制御ビットの挿入では、次の「# 2 cell」のデータ列の変換(入れ替え)は行わず、区間DSVが計算される。そして積算DSV値より、図2(C)中の「d1」のDSV制御ビットが決定される。

【0087】例えば図2(D)のような場合、データ配列変換は、各「cell」の最後に、IDビットとDSV制御ビットを挟む。IDビットは、その「cell」の先頭同期信号が複数種類あるときの、識別用の情報ビットであり、各「cell」の最後においてDSV制御ビットを挟むように、その前に配置される。この方式の場合は、SYNC/SyncID挿入部22は、同期信号(24ビット)のみを挿入し、またSyncIDの決定は、実質的には、データ配列変換部12において行われることになる。

【0088】以上のようなデータ配列変換を行った2種類のデータ列の変調は、次の変調部13で行われる。変調部23は、変換テーブルを内蔵している。さらにSYNC/SyncID挿入部22は、それぞれ所定の間隔で変調された信号に同期信号を挟む。SYNC/SyncID挿入部22は、終端テーブルを内蔵し、同期信号を挟むために終端されたデータ列を、符号語列に変換する。その符号語列は、NRZI化部16でレベル符号化される。この時点で、チャンネルビット列は、まだそのDSV制御ビットが決定されておらず、2種類のレベル符号列が存在する。そしてDSVビットSYNC決定部23は、それぞれDSV値を計算し、積算されたDSVが抑制される方のどちらかのチャンネルビット列を選択し、これを決定する。ここで同時に同期信号のパターンが決定されることになる。決定された符号語列(チャンネルビット列)は、DSV制御が行われたデータ列として出力される。

【0089】続いて、本発明に係る復調装置の一実施の形態を図面を参照しながら説明する。この実施の形態は、データ列を表3の変長符号(d, k; m, n; r) = (1, 7; 2, 3; 4)に変換した変調符号語列を復調する復調装置に適用したものである。

【0090】図6は同期信号を含み、またデータ配列変換が行われた記録符号列の再生データを復調する復調装置の構成を示すブロック図である。コンパレート/逆NRZI化部31は、伝送路より伝送されてきた信号、また

は、記録媒体より再生された信号をコンパレートし、逆NRZI化し(エッジ符号にし)、その結果を、復調部32およびSYNC/SyncID識別部33に供給する。復調部32は、エッジ符号化されたデジタル信号を復調テーブル(逆変換テーブル)に基づいて復調し、SYNCビット取出部34に出力する。SYNC/SyncID識別部33は、所定の間隔で挿入されている同期信号(Sync)を識別し、同期信号部分の直前において終端テーブルの逆変換終端テーブルが用いられている場合、その情報を復調部32に送り、また、同期信号の後ろの6ビットよりSyncIDを識別する。

【0091】SYNCビット取出部34は、同期信号を取り出す。データ配列変換部35は、データ列の配列変換が行われた部分を逆変換して元に戻す。DSVビット取出部36は、復調されたデータ列より、任意の間隔で挿入されているデータ列内のDSV制御ビットを取り除き、元のデータ列を出力する。バッファ37は、DSVビット取出部36から入力されたシリアルデータを一旦記憶し、所定の転送レートで読み出し、出力する。タイミング管理部38は、タイミング信号を生成し、コンパレート/逆NRZI化部31、復調部32、SYNC/SyncID識別部33、SYNCビット取出部34、データ配列変換部35、DSVビット取出部36、およびバッファ37に供給し、タイミングを管理する。

【0092】SYNC/SyncID識別部33は、固有のパターンによって同期信号の位置を決定すると共に、所定の間隔で同期信号が入っているのをカウントすることによってその位置を定めることが出来る。同期信号の位置が判明したとき、その直前付近の復調は、終端テーブルを含めて行われる。一方、同期信号の直後において、終端テーブルは不要であり、表3の通常テーブルで復調ができる。

【0093】SYNCビット取出部34は、上に説明したようにして直前の復調が行われた後、所定の同期信号のビット数だけ取り除き、復調部32と整合性を取る。

【0094】データ配列変換部35は、図2(B)乃至図2(D)に示す形式のように、変調時にデータ配列変換されたものを逆変換する。

【0095】次に図6の復調装置の動作について説明する。

【0096】伝送路より伝送されてきた信号、または記憶媒体より再生された信号は、コンパレート/逆NRZI化部31に入力され、コンパレートされるとともに、逆NRZI符号("1"がエッジを示す符号)のデジタル信号に変換され、復調部32およびSYNC/SyncID識別部33に入力される。

【0097】このデジタル信号は、復調部32において、表3の逆変換テーブルに基づいて復調される。復調部32は、表3の逆変換テーブルを有するが、終端用の逆変換テーブルは必ずしも持たなくてもよい。その場

合、同期信号が挿入された直前部分で逆変換が不可能となるときがあるが、このときはSYNC/SyncID識別部33において、これを捕う。SYNC/SyncID識別部33は、同期信号の検出情報を送り、復調部32は、これに同期して復調を開始する。

【0098】SYNC/SyncID識別部33は、同期信号のパターンとして与えられた部分の、2T-9T-9Tの部分を示す、"x0101000000001000000010"を検出する。この同期信号のパターンは固有なパターンである9Tを含むので、他の情報符号語列内からは、検出されることはない。またSYNC/SyncID識別部33は、一度同期信号のパターンが検出されたら、それ以降は内部カウンタ等によって、所定の間隔の同期信号のパターンを検出することが出来る。

【0099】SYNC/SyncID識別部33は、また、終端テーブルの逆変換テーブルも有し、同期信号の直前において、終端のために用いられた終端テーブルによって作られた符号語を復調し、その結果を復調部32に送る。結局、終端の逆変換テーブルは、復調部32またはSYNC/SyncID識別部33のいずれかが持てばよい。

【0100】SYNC/SyncID識別部33は、さらに、同期信号のパターンである2T-9T-9Tの後ろに続く、複数種類の同期信号を識別する。それぞれの同期信号は、例えば検出能力が強くされたパターンが選択されて

<表5>

逆変換テーブル

PP17- (d,k;m,n;r) = (1,7;2,3;4) r=4,

符号語列 復調データ列

i=1	101	11
	000	11
	001	10
	010	01
i=2	010 100	0011
	010 000(not 100)	0010
	000 100	0001
i=3	000 100 100	000011
	000 100 000(not 100)	000010
	010 100 100	000001
	010 100 000(not 100)	000000
i=3	Prohibit Repeated Minimum Transition Runlength	
	001 000 000(not 100)	110111
i=4	limits k to 7	
	000 100 100 100	00001000
	010 100 100 100	00000000

【0106】<表6>

逆変換テーブル

Termination table

000 00

いる。

【0101】同期信号の30ビットは、SYNCビット取出部34において取り除かれる。データ配列変換部35は、図2(B)または、図2(C)に示す形式のように、変調時のデータ配列変換の逆変換を実行する。すなわち図2(B)の場合、データ配列変換部35は、「#1 cell」の最後でDSV制御ビット「d0」を検出し、その結果、「#0 cell」の最後のデータ「23」の反転/非反転を決定し、実行する。また、図2(C)の場合、データ配列変換部35は、「#1 cell」の先頭と最後のデータ「45」乃至「23」を逆に並べ替える。このような動作が繰り返される。

【0102】データの配列が図2(D)に示す形式の場合、同期信号ビットは24ビットであり、またSyncIDビットは各「cell」の後方に挿入されているところが異なるが、その他は同様に処理される。

【0103】そして、DSVビット取出部36は、さらに、所定の間隔で挿入されているDSV制御ビットを取り除く。

【0104】表3の変換テーブルを使用した処理で得られた符号列を復調するときに使用する逆変換テーブルは、例えば、次の表5のようになる。また終端の逆変換テーブルは、例えば、次の表6のようになる。

【0105】

010 100 0000

【0107】本発明の効果をシミュレーションにより示す。シミュレーションは、図2(A)に示した符号列に対する、図2(B)および(C)に示した符号列の比較

である。図2 (D) については、冗長度の差があるためにシミュレーションは省略したが、2ビット単位の規則の関係は図2 (B) および (C) と同様にずれが存在しないことより、図2 (D) の符号列のシミュレーション結果は、図2 (B) および (C) に示した符号列と同等であると考えられる。

【0108】まず、シミュレーションによるエラーレートの測定方法について説明する。表3の変換テーブルに

<表7>

<<Byte エラーレートシミュレーション結果>>

	図2 (A)	図2 (B)	図2 (C)
付加エラー数	100000	100000	100000
1Byte連続エラー	64489	68857	70133
2Byte連続エラー	19482	16424	15857
3Byte連続エラー	35	21	17
4Byte連続エラー	0	0	0
totalエラー箇所	84006	85302	86007
totalエラー数	101898	103558	101768
平均Byte-error prop.	1.0177 Byte	1.0356 Byte	1.0177 Byte

【0110】totalエラー箇所が、付加エラー数と等しくならないのは、Sync・IDに含まれるエラーや、あるいは復調の結果、エラーを起こさないでデータ復調が出来たことを示している。図2 (B) および、図2 (C) の符号列の復調では、2Byte連続エラー、3Byte連続エラーは減少し、1Byteエラーが増加した。totalエラー数は、図2 (B) の符号列が少なく、次が図2 (C) の符号列であり、図2 (A) の符号列が最も多かった。2Byte連続エラーについて、図2 (B) の方が15%程度エラーが少なくなるが、totalエラー箇所は図2 (A) に比べ、1.5%程度増加する。この中には、図2 (B) および図2 (c) の符号列の特徴である、長いバイトエラーが分散して、1ヶ所の1ビットシフトエラーが離れた2ヶ所にエラーとして出てしまうことが影響しているのではないかと考えられる。すなわち、変復調前後（ディスクへの記録再生前後）にビットの入れ替えをするというこ

【0111】表3の変調テーブルより、

10<->001

01<->010

というように、1ビットシフトのエラーが、di-bitの2ビット共を反転させてしまう変復調があり、このdi-bitがバイトとバイトの間にまたがっていると2バイトエラーになってしまうので、このdi-bitをバイト間にまたがら

よる変調の後、チャンネルビット列の所定個目の"1"毎に、その"1"を前後に1ビットずらしてシフトエラーを起こさせた符号列を復調し、これを変調前のデータ列と比較して何バイトエラーになるかを調べた。エラーとエラーの間隔は、隣同士に発生させたエラーが干渉し合わないように選択した。エラー発生回数は、それぞれ同数としてある。

【0109】

せないようにして2バイトエラーを減らした。その結果、図2 (B) 、および図2 (C) の2バイト、3バイトエラーは減っているが、一方でエラーを発生した箇所は増加した。

【0112】まとめると、データ配列変換を行うことによって、シフトエラー時のエラー伝搬のうち、2バイトや3バイトといった長いエラー伝搬を減らすことができることがわかった。さらに、totalエラー数も減らし、平均Byte-error伝搬も減らすことが出来ることがわかった。

【0113】ゆえに、データ配列変換によって、シフトエラー時の長いエラー伝搬の発生を少なくし、平均バイトエラー伝搬を少なくすることが出来るので、より安定な復調が可能になる。

【0114】なお、本明細書において、システムとは、複数の装置により構成される装置全体を表すものとする。

【0115】なお、上記したような処理を行うコンピュータプログラムをユーザに提供する提供媒体としては、磁気ディスク、CD-ROM、固体メモリなどの記録媒体の他、ネットワーク、衛星などの通信媒体を利用することができる。

【0116】

【発明の効果】請求項1に記載の変調装置、請求項5に記載の変調方法、および請求項6に記載の提供媒体によれば、データ列に1ブロック当たり1ビットのDSV制御

ビットを挿入し、データ列の、同一のバイト単位に属するデータが、同一の変調単位となるようDSV制御ビットが挿入されたデータ列を並び替えるようにしたので、復調時の長いエラー伝搬を減らし、より確実なデータの再生ができるようにすることができる。

【0117】請求項7に記載の復調装置、請求項8に記載の復調方法、および請求項9に記載の提供媒体によれば、変調時のデータ列の並び替えに対応して、復調されたデータ列を並び替えるにしたので、復調時の長いエラー伝搬を減らし、より確実なデータの再生ができるよう

【図面の簡単な説明】

【図1】記録符号列フォーマットの一例を表す図である。

【図2】ビット配列の変換例を示した図である。

【図3】記録符号列フォーマットの一例を表す図である。

る。

【図4】変調装置の一実施の形態の構成を示すブロック図である。

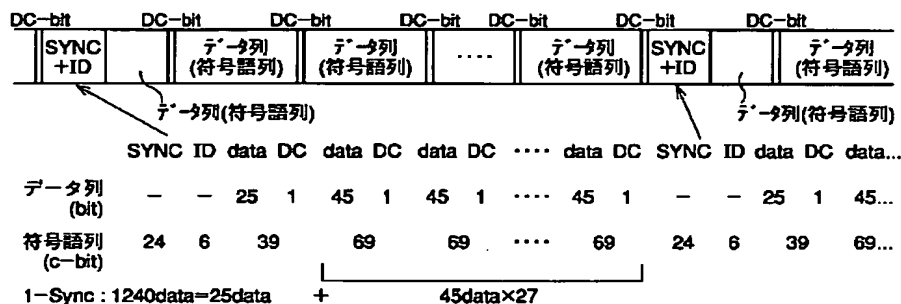
【図5】変調装置の他の実施の形態の構成を示すブロック図である。

【図6】復調装置の一実施の形態の構成を示すブロック図である。

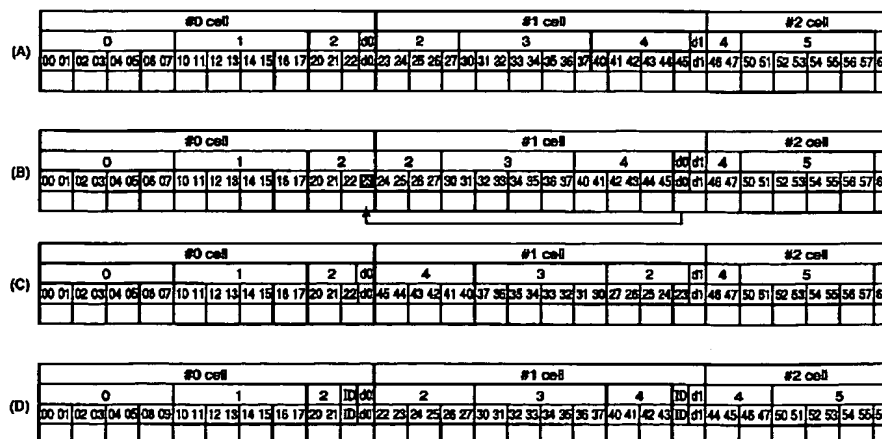
【符号の説明】

11 DSVビット決定・挿入部, 12 データ配列変換部, 13 変調部, 14 SYNC/SyncID決定部, 15 SYNCビット挿入部, 16 NRZI化部, 21 コントロールビット挿入部, 22 SYNC/SyncID挿入部, 23 DSVビットSYNC決定部, 31 コンパレート/逆NRZI化部, 32 復調部, 33 SYNC/SyncID識別部, 34 SYNCビット取出部, 35 データ配列変換部, 36 DSVビット取出部

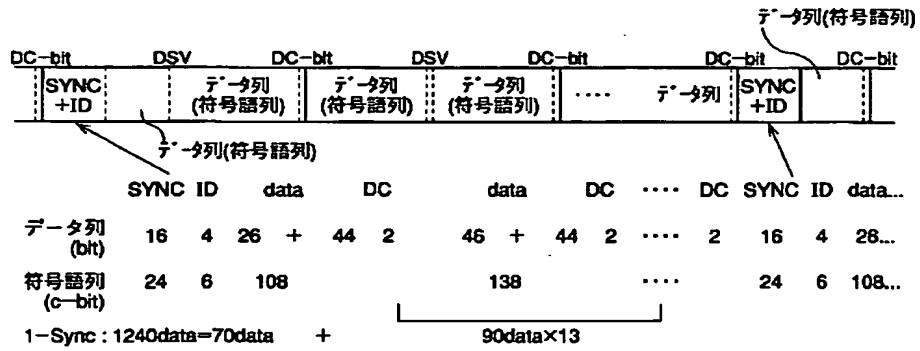
【図1】



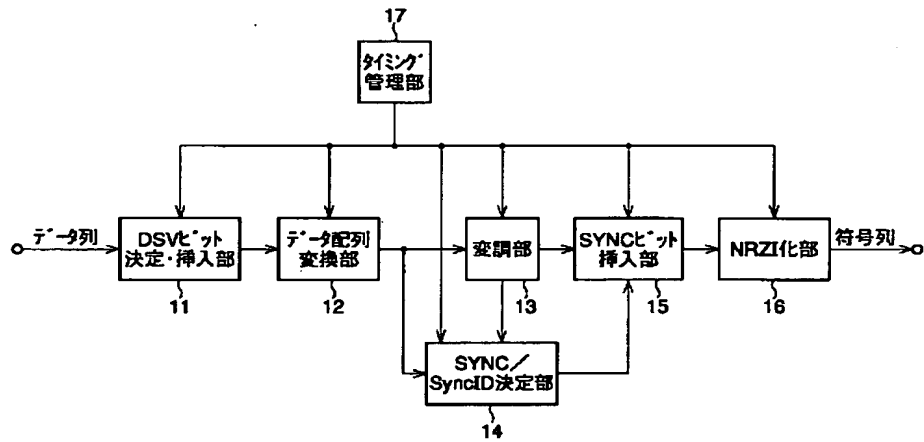
【図2】



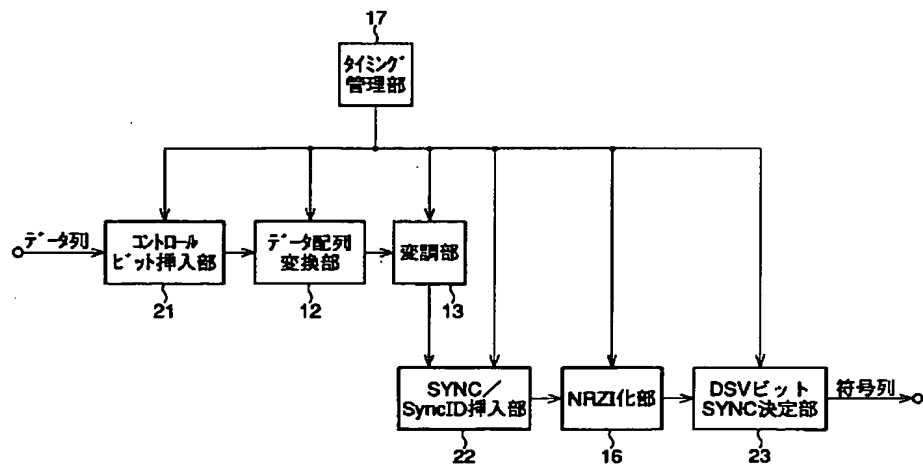
【図 3】



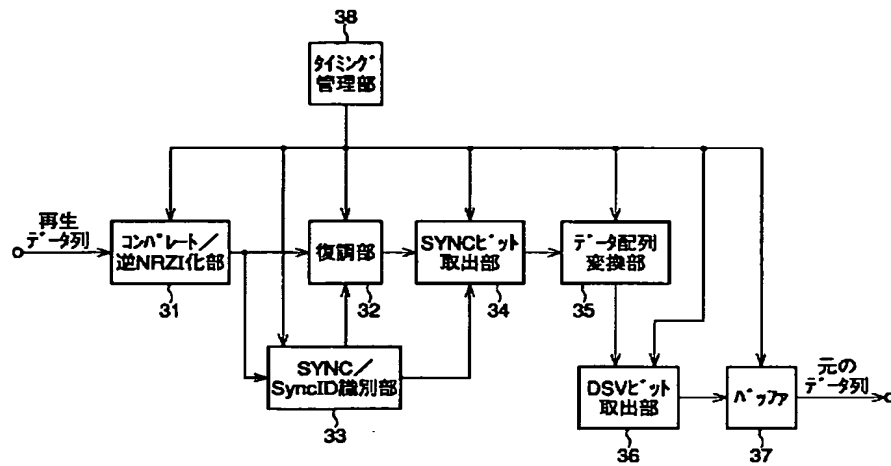
【図 4】



【図 5】



【図 6】



フロントページの続き

- (72)発明者 榎原 立也  
 東京都品川区北品川 6 丁目 7 番 35 号 ソニ  
 ー株式会社内
- (72)発明者 中村 耕介  
 東京都品川区北品川 6 丁目 7 番 35 号 ソニ  
 ー株式会社内